

Linux for Cell Broadband Engine and PS3, Related Open Source Projects

Nov 2nd, 2007

Hiroyuki Machida

What's Cell Broadband Engine (Cell/B.E.)

Contents

- * What's Cell Broadband Engine
 - * Status of Linux Distro
 - * Cell/B.E. Performance
 - * Hello SPE
 - * Various OSS activities
-

Cell/B.E. - A Heart of PS3

One PPE (SMT)

Six SPE

- * One: reserved for System Software
- Another: turned off for better yield rate



```
sdf: Write Protect is off
sdf: assuming drive cache: write through
sdf: sdf1
sd 4:0:0:0: Attached scsi removable disk sdf
Loading usbhid.ko module
input: Dell Dell USB Keyboard as /class/input/input0
input: USB HID v1.10 Keyboard [Dell Dell USB Keyboard] on usb-0000:00:01.1-2.4
usbcore: registered new driver usbhid
drivers/usb/input/hid-core.c: v2.6:USB HID core driver
Creating root device.
Mounting root filesystem.
kjournald starting. Commit interval 5 seconds
EXT3-fs: mounted filesystem with ordered data mode.
Setting up other filesystems.
Setting up new root fs
no fstab.sys, mounting internal defaults
Switching to new root and running init.
umounting old /dev
umounting old /proc
umounting old /sys
INIT: version 2.86 booting
```

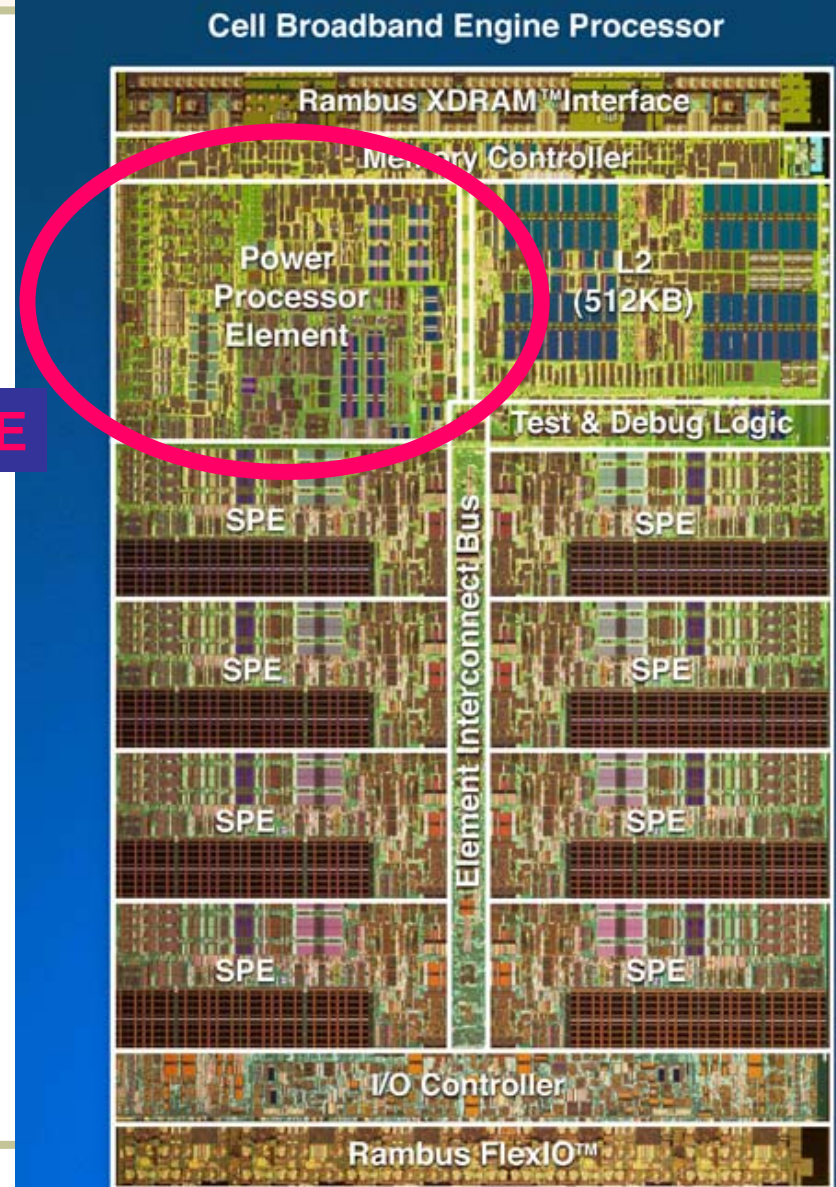
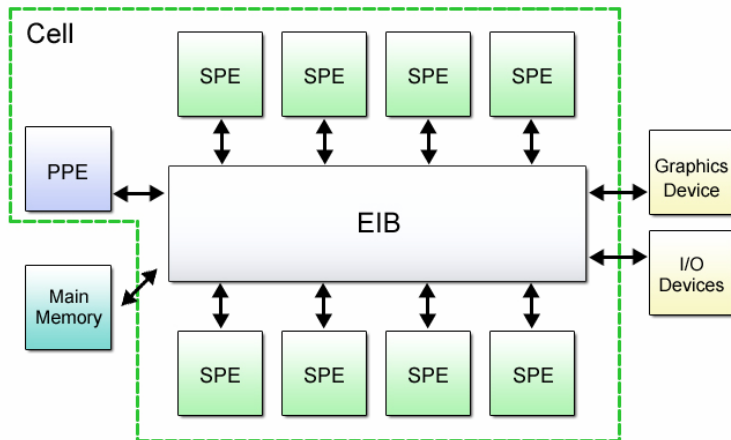


* Heterogeneous Multi-core Processor

- 1 PPE (Power Processor Element)
 - * PPU + 512KB L2 Cache
 - * PPU → PPC64 insn + VMX instructions, SMT/in-order
 - 8 SPE (Synergetic Processor Element)
 - * Generic 4way SIMD Processor (SPU, incompat with PPC)
- + LS_(Local Storage) + MFC_(Mem. Flow Cntl)

SPEs – Key of High Performance

- * Design Strategies
 - “simpler structure and higher clock”
 - “more room for SPEs on silicon”
- * PPE does NOT achieve same speed with same clock G5 (PPC970)



SPE Memory Architecture

- * Like no other ...
 - Small (256KB) and High speed LS, instead of Cache
 - * LS stands for Local Store
 - Intelligent DMA (MFC, EA based, coherent check)
 - * EA stands for Virtual Address Space



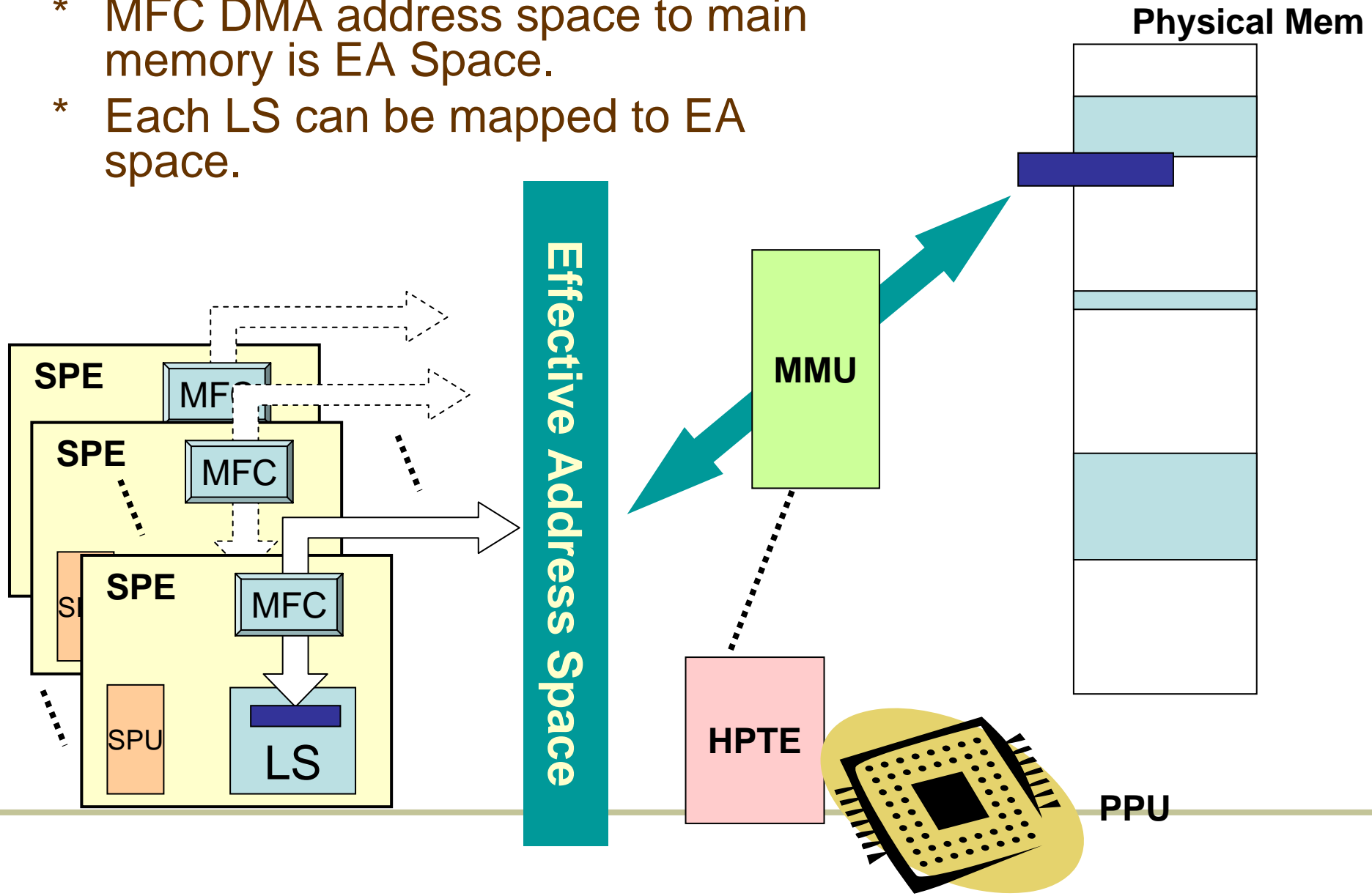
Able to control precise timing and amount of DMA
→ Peak Performance



Need to address what cache did
→ More efforts

DMA issued by SPE (MFC)

- * MFC DMA address space to main memory is EA Space.
- * Each LS can be mapped to EA space.



SPE Memory Architecture

Embedded Engineers and Kernel Hackers must like this

- Small (256KB) and High speed LS, instead of Cache
 - * LS stands for Local Store
- Intelligent DMA (MFC, EA based, Coherent Check)
 - * EA stands for Virtual Address Space



Able to control precise timing and amount of DMA
→ Peak Performance



Need to address what cache did
→ More efforts

Status of Linux Distro for Cell/B.E. and PS3



Yellow Dog Linux

* Nov, 2006
- YDL 5.0

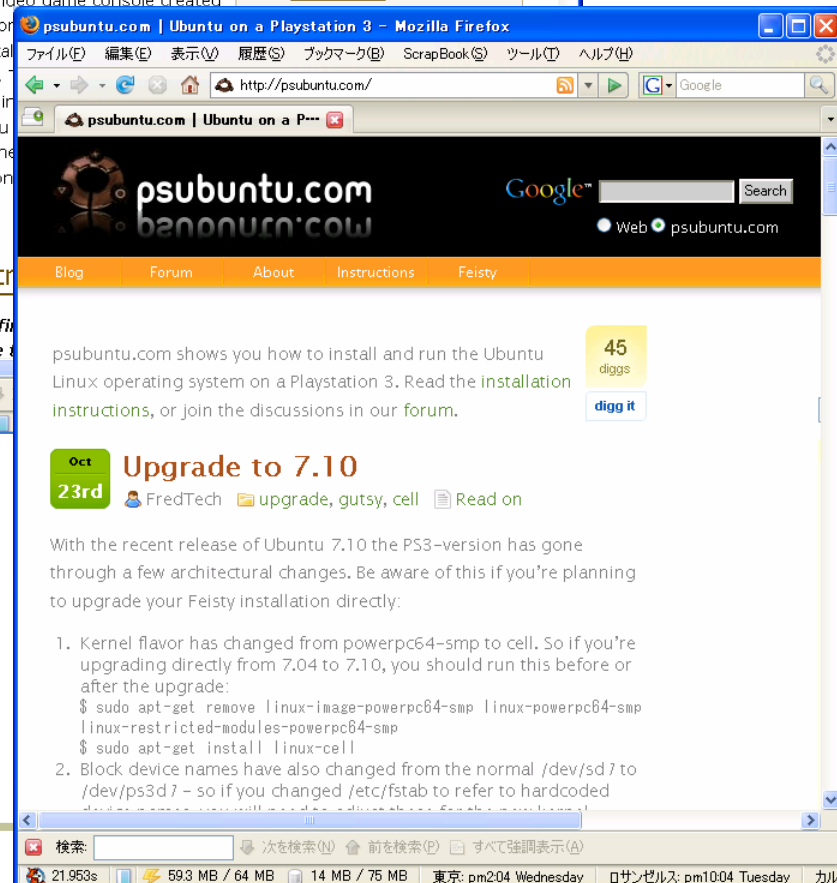
* Jun, 2007
- YDL 5.0.2





* April, 2007
- Ubuntu 7.04

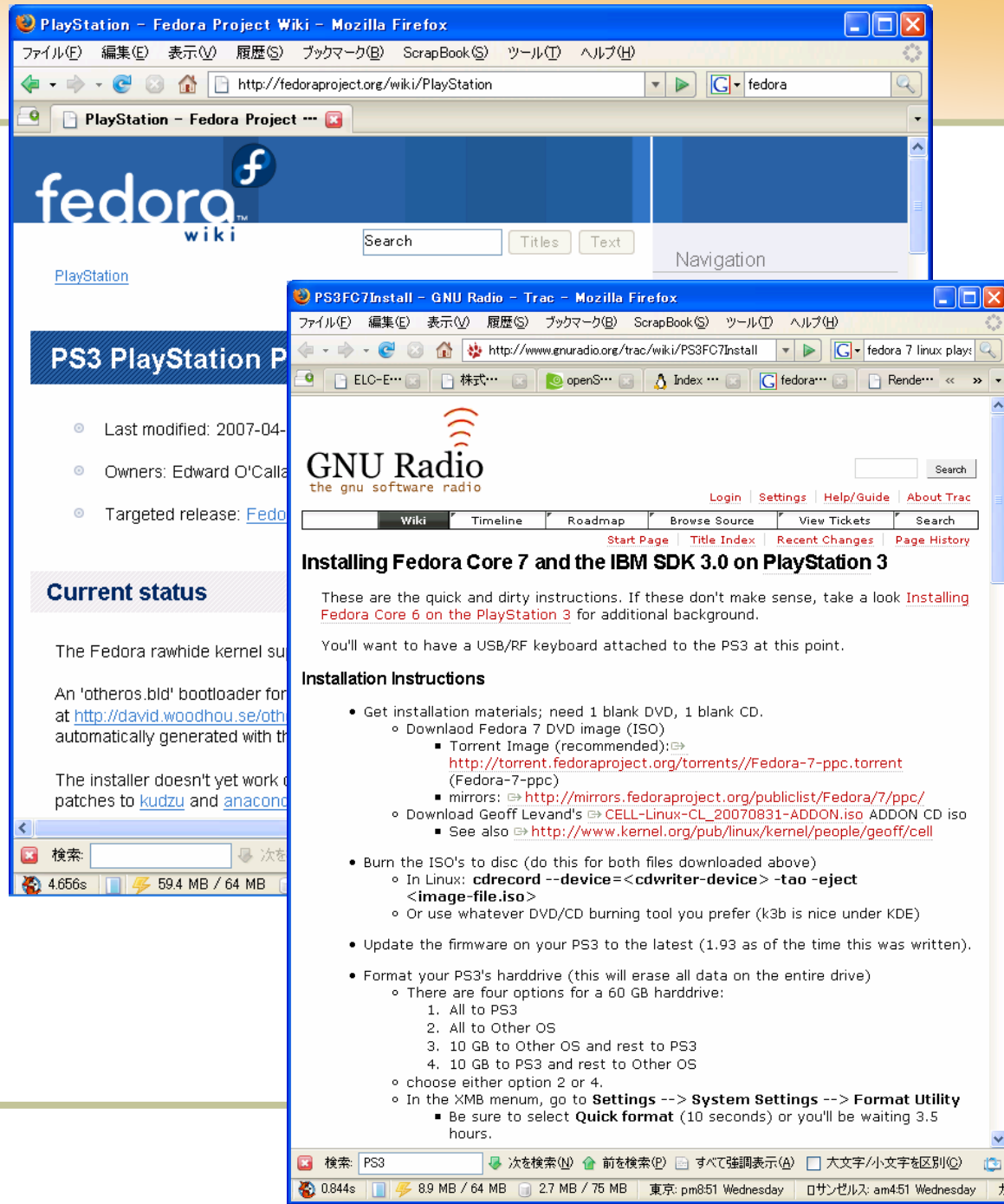
* Oct, 2007
- Ubuntu 7.10



fedora

* May, 2007
- Fedora 7

* Nov, 2007
- Fedora 8 –
Under
Development





* Oct, 2007
- Open Suse 10.3

The screenshot shows a Mozilla Firefox browser window with the address bar displaying <http://en.opensuse.org/PS3>. The page title is "PS3 - openSUSE". The main content area features a "Discover it" section with a search bar and a "PS3" article. The article title is "CURRENT STATUS of openSUSE on PS3: Working". The article text states: "If you have any info please help by updating this page".

Below the article, there is a section titled "Linux and Open Source Blog" with a date of "October 6, 2007". The blog post is titled "OpenSUSE 10.3 On PS3" and is filed under "1. openSUSE/SUSE, Games & Gaming, HowTos/Tutorials/Tips" by "E@zyVG" at "1:48 pm". The post content reads: "The [spicy power team](#) concentrated for openSUSE 10.3 on supporting the Sony Playstation PS3, the major changes are:

- openSUSE 10.3 is fully installable with YaST on the Sony Playstation PS3
- a kexec-based bootloader called [petitboot](#) is used for PS3
- There is a development pattern for Cell development on PS3 to easily install the needed packages

More details on installing openSUSE 10.3 on PS3 can be found at <http://en.opensuse.org/PS3>.

Besides the PS3, 10.3 also runs on the previously supported Apple Power Mac, IBM POWER, and Pegasos PPC hardware for details see <http://en.opensuse.org/POWER@SUSE>.

The blog post has "2 Comments":

1. wow, great article. Although i haven't seen someone did this before. Well, if i was in front of my PlayStation, then it means i want to play Final Fantasy or Tekken, not Linux anymore. 😊 BTW, did the mouse and keyboard is a regular mouse and keyboard or special for PS3 only....
2. [...] unknown wrote an interesting post today onHere's a quick excerptopenSUSE 10.3 is fully installable with YaST on the Sony

The comment by "linux newbie" is dated "October 6, 2007 @ 2:19 pm".

The right sidebar of the blog post contains a section titled "the linux & open source blog" with a description: "this site is dedicated to news, reviews, thoughts and trends in linux and open source world." Below this is a "pages" section with links to "1. about me", "2. oss/linux", "3. my system", "4. links", "4a. thinking of using linux?", "4b. welcome windows users", "4c. linux is not windows", "4d. getting suse help", "4e. sled 10 installation", "4f. openoffice.org writer", "4g. open source apps on mac". There is also a "poll-booth" section with the text "prefered distro poll has been ended. soon a new one, -----" and a "Search" bar. The bottom of the sidebar has a "topics & categories" section with links to "applications/software (446)", "browsers (82)", and "business (9)".

The browser window also shows a status bar at the bottom with the text "8.470s 9.9 MB / 64 MB 3 MB / 75 MB 東京: pm8:55 Wednesday ロサンゼルス: am4:55 Wednesday カルカッタ: pm5:25".

And More...



YDL



Gentoo



ubuntu

fedora^f



debian



Momonga
Linux



OPEN SUSE™

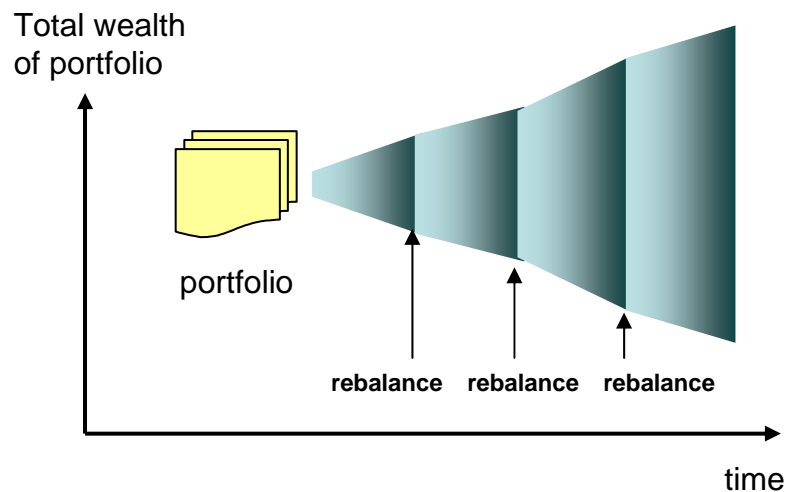
You can install your favorite Distro !

Cell/B.E. Performance

What can you do with SPEs

Examples - Finance & Recognition

Multi-period Stochastic Portfolio Optimizer



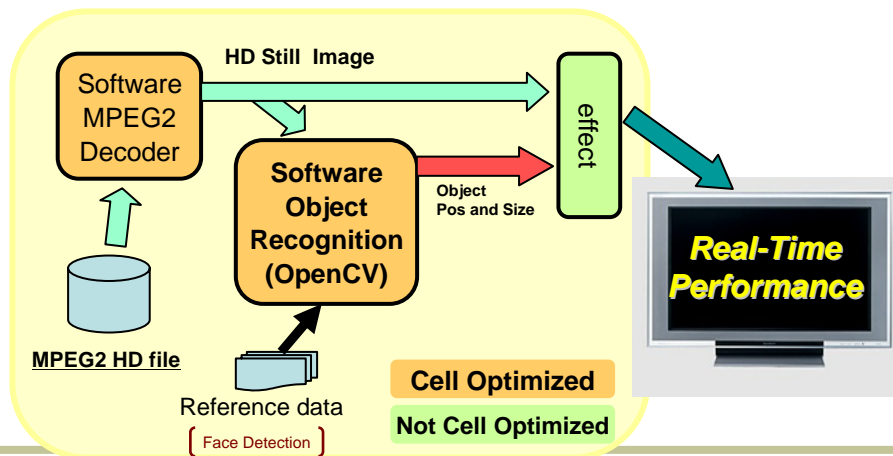
Performance of Important subroutine (Cholesky factorization)

GFLOPS

Cell/B.E. 3.2GHz	175 (8SPE)
Core2Duo 2.6GHz	33 (2Core)

5 times faster!!

Real-Time Object Recognition with MPEG2 HD Stream



Object Recognition Speed

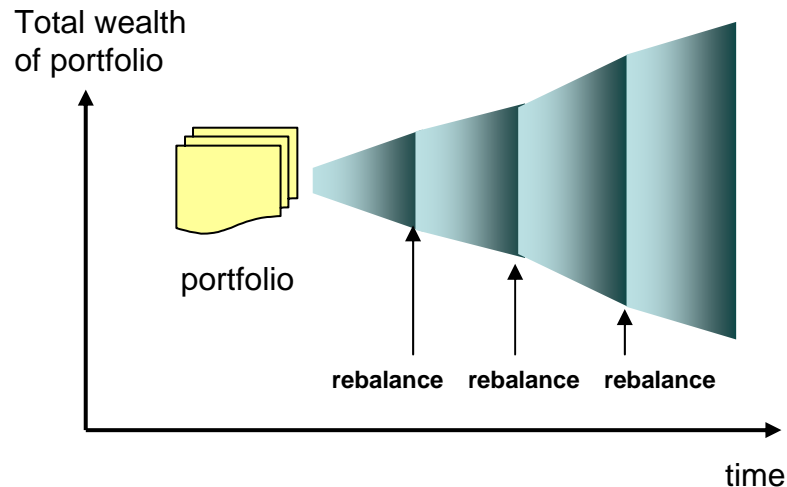
msec/frame

Cell/B.E. 3.2GHz	27.1 (4SPE)
Core2Duo 3.0GHz	57.9 (2Core)

2 times faster!!

Some Trial to Finance

- * Multi-period portfolio management theory
 - calculates the optimal portfolio in every future periods
 - formulated as a large scale liner programming



Minimize $\frac{1}{I} \sum_{i=1}^I q^{(i)}$

subject to

$$\sum_{j=1}^n \rho_{j0} z_{j0} + v_0 = W_0$$

$$\sum_{j=1}^n \rho_{j1}^{(i)} z_{j1} + v_1^{(i)} = \sum_{j=1}^n \rho_{j1}^{(i)} z_{j0} + (1 + r_0) v_0 \quad (= W_1^{(i)}), \quad (i = 1, \dots, I)$$

$$\sum_{j=1}^n \rho_{jt}^{(i)} z_{jt} + v_t^{(i)} = \sum_{j=1}^n \rho_{jt}^{(i)} z_{j,t-1} + (1 + r_{t-1}^{(i)}) v_{t-1}^{(i)} \quad (= W_t^{(i)}),$$

($t = 2, \dots, T-1; i = 1, \dots, I$)

$$\sum_{j=1}^n \bar{\rho}_{jT} z_{jT-1} + \frac{1}{I} \sum_{i=1}^I (1 + r_{T-1}^{(i)}) v_{T-1}^{(i)} \geq W_E$$

$$\left\{ \sum_{j=1}^n \rho_{jT}^{(i)} z_{jT-1} + (1 + r_{T-1}^{(i)}) v_{T-1}^{(i)} \right\} + q^{(i)} \geq W_G, \quad (i = 1, \dots, I)$$

$$z_{jt} \geq 0, \quad (j = 1, \dots, n; t = 0, \dots, T-1)$$

$$v_0 \geq 0$$

$$v_t^{(i)} \geq 0, \quad (t = 1, \dots, T-1; i = 1, \dots, I)$$

$$q^{(i)} \geq 0, \quad (i = 1, \dots, I)$$

Many
valuables

Many
equations



Power of Cell Broadband Engine

Hiroyuki Machida / Sony Corp.

What is demonstrated

Real-Time Object Recognition with MPEG2 HD Stream

As an example of a High Performance Application enabled by the Cell Broadband Engine (Cell/B.E.), we demonstrate Real-Time Object Recognition with a MPEG2 HD stream, implemented using a version of OpenCV optimized for the Cell/B.E.

For reference, we also show the same application running on a conventional PC equipped with the latest multi-core CPU. The PC cannot achieve such a real-time performance as the Cell/B.E..

Object Recognition Speed

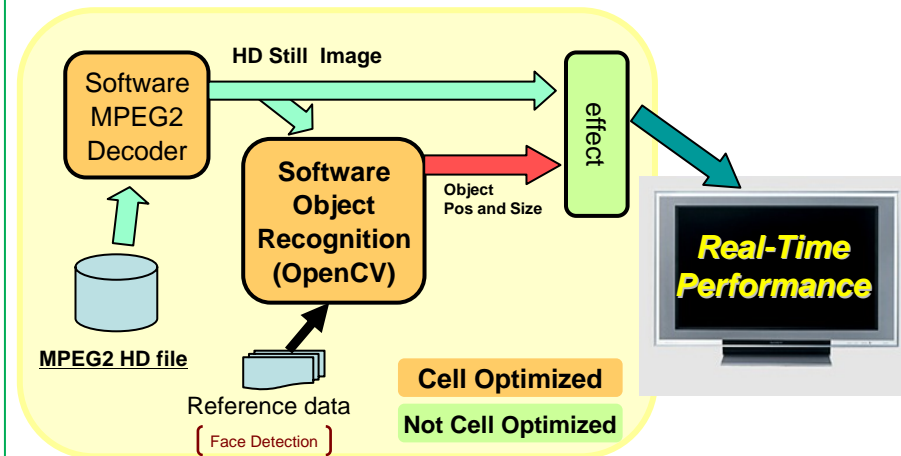
msec/frame

Cell/B.E. 3.2GHz	27.1 (4SPE)
Core2Duo 3.0GHz	57.9 (2Core)

Hardware Information

PLAYSTATION®3

How it was improved



Functions in OpenCV have been off-loaded to multi-core SIMD processors(SPE) inside Cell/B.E.

Patch or technical information availability

Cell/B.E. optimized OpenCV Project

http://cell.fixstars.com/opencv/index.php/OpenCV_on_the_Cell

Cell/B.E. optimized OpenCV Patches

<https://sourceforge.net/projects/cvcell/>

OpenCV Project

<http://opencvlibrary.sourceforge.net/>

Hello SPE

An Introduction to SPE Programming

SPE Programming Environment

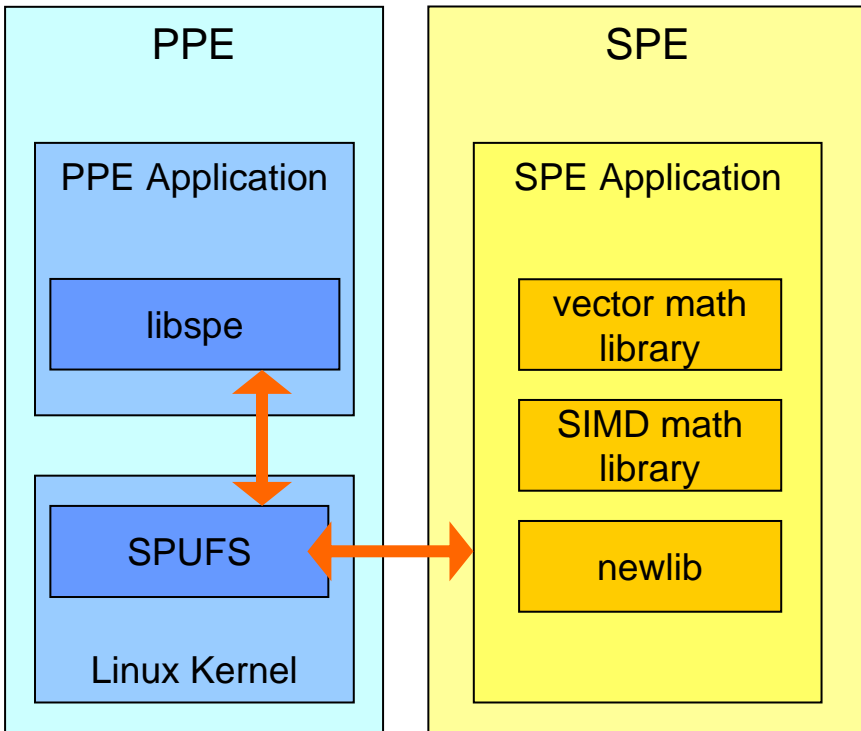
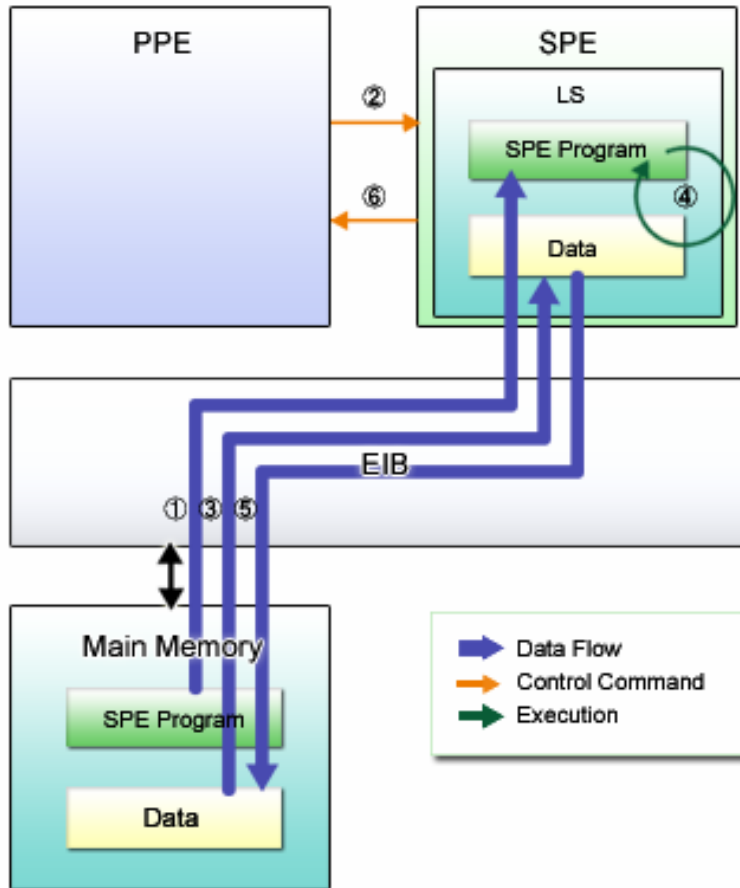


Diagram of SPE runtime environment

- * SPE management library - libspe2
- * SPU toolchain - spu-gcc, ...
 - GCC, BINUTILS
- * libc for SPE - newlib
 - SPE Optimized strings/mem functions, PPE offloading , ...
- * SPE elf program launcher - elfspe
- * Combined GDB – ppu-gdb
 - both PPU and SPU programs.
- * PPU optimized toolchain (recommended)
 - ppu-gcc, ...
 - GCC, BINUTILS

Typical Cell/B.E. Program Execution Flow



* PPE Side

- `spe_create_context()`
- `spe_image_open()`
- [1] `spe_program_load()`
- [2] `spe_context_run()`

* SPE Side

- [3] Data: Main Memory → LS
- [4] Process data in LS
- [5] Data: LS → Main Memory
- [6] Signal to PPE program

Hello SPE

- * *elfspe* allows small stand alone SPE program run directly

```
% cat hello-spe.c

#include <stdio.h>

int main(unsigned long long spe, unsigned long long argp, unsigned long long envp)
{
    printf("Hello SPE!¥n");
    return 0;
}

% spu-gcc -Wall -Os -ffast-math -ftree-vectorize \
  -ffunction-sections -fdata-sections -Wl,-gc-sections \
  -o hello-spe.elf hello-spe.c

% ./hello-spe.elf
Hello SPE!

%
```

- * For more details, please see “[Cell Programming Primer](#)” in PS3 Linux Distributor's Starter Kit

PS3 Linux Documentation

- * Linux Kernel Overview
- * How to Enable Your Distro
- * Booting Linux and Installation
- * Platform Specific Utilities
- * Application Programming Environment
- * Open Source Communities
- * **Cell Programming Primer**

It's a entry-level Cell/B.E. programming tutorial

Basics of SPE Programming - Mozilla Firefox

http://www.kernel.org/pub/linux/kernel/people/geoff/cell/CELL-Linux-CL_20070516-ADDON/doc/CellProgrammingTutorial/E

```
51  /* DMA Transfer 3 : PUT output data */
52  spu_mfcdma64(vout, mfc_ea2h(abs_params.ea_out), mfc_ea2l(abs_params.ea_out),
53              abs_params.size * sizeof(float), tag, MFC_PUT_CMD);
54  spu_writet(MFC_WrTagMask, 1 << tag);
55  spu_mfcstat(MFC_TAG_UPDATE_ALL);
56
57  return 0;
58 }
```

Lines 45~49 Used to calculate absolute values by SPU SIMD operations.

[Click here for the source code for absolute value calculations by SPU SIMD operations.](#)

3.5 Parallel Programming Using Multiple SPEs

All sample programs presented so far in this chapter run on a single SPE.
Finally, we will now take a look at the method to execute an application using multiple SPEs.

3.5.1 Data Partitioning

With Cell programming, how the work is partitioned among available SPEs is an important consideration. Although there are many ways of doing this, the following explanation is based on a model that subdivides data to enable concurrent processing.

The data-partitioning application model parallels the same program across multiple SPEs. Data is partitioned by the PPE program and uniformly distributed to SPEs. Fig. 3.6 provides an image of this approach.

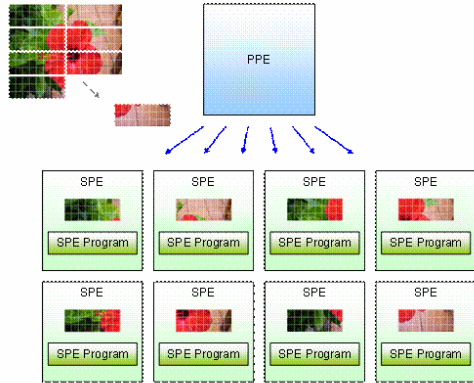


Fig. 3.6 : Data Partitioning for Parallel Processing

Programming is performed as follows.

3.5.2 Absolute Value Calculation Using Multiple SPEs

Example of absolute value calculation. In Sections 3.3 and 3.4, negative integers contained in the 64 data inputs are converted into positive integers.

63 MB / 64 MB 19 MB / 75 MB 東京 pm9:41 Friday ロサンゼルス am5:41 Friday カルカッタ pm6:11 Friday プリスベン pm10:41 Friday ベルリン pm2:41 Friday

Various OSS activities

Linux Distribution Starter's Kit

- * Latest Releases
 - **V1.4.1** **Aug 2007** **Geoff's Kernel.Org Git Tree Kernel**
 - **V1.5** **Oct 2007** **Performance Monitor Support**
- * Purpose
 - Providing technical information to create Linux distributions on PS3
- * **Contents**
 - PS3 Linux Documentation
 - PS3 Linux kernel source
 - PS3 Platform utilities with source
 - PS3 framebuffer sample programs
 - PS3 Linux boot loader (kboot)
- * CD-image
 - <ftp://ftp.uk.linux.org/pub/linux/Sony-PS3>
 - <ftp://ftp.infradead.org/pub/Sony-PS3>
- * Extracted (v1.5)
 - http://www.kernel.org/pub/linux/kernel/people/geoff/cell/CELL-Linux-CL_20071023-ADDON

Contributions to The Latest Kernel

Who wrote 2.6.23 [LWN.net] - Mozilla Firefox

http://lwn.net/Articles/247582/

supported this work, with the results shown below. These results should always be taken as approximations; we believe that they are essentially correct, but patches do not come with Paid-for-by: headers, so a certain amount of guessing is always required.

Most active 2.6.23 employers

By changesets			By lines changed		
(Unknown)	1180	19.0%	(Unknown)	111777	16.9%
Red Hat	744	12.0%	(None)	99649	15.0%
(None)	559	9.0%	Red Hat	84224	12.7%
IBM	507	8.2%	IBM	39449	5.9%
Novell	421	6.8%	Oracle	36205	5.5%
Intel	184	3.0%	Renesas Technology	33152	5.0%
Oracle	146	2.4%	HP	18718	2.8%
Renesas Technology	134	2.2%	Tripeaks	18567	2.8%
MIPS Technologies	119	1.9%	Novell	17990	2.7%
NetApp	116	1.9%	Emulex	15942	2.4%
(Consultant)	103	1.7%	XenSource	15426	2.3%
Google	99	1.6%	Intel	14962	2.3%
NTT	98	1.6%	Sony	11945	1.8%
Sony	93	1.5%	Analog Devices	10345	1.6%
Astaro	93	1.5%	rPath	9678	1.5%
Linux Foundation	82	1.3%	MIPS Technologies	9171	1.4%
MontaVista	81	1.3%	Solid Boot Ltd.	8937	1.3%
SGI	77	1.2%	MontaVista	8065	1.2%
Qumranet	72	1.2%	PMC-Sierra	6888	1.0%
QLogic	62	1.0%	Astaro	6687	1.0%

Red Hat retains its place at the top of the by-changesets list, though its percentage of changes has dropped a bit. By lines changed, developers known to be working on their own time (the "None" entry) beat out all corporate contributors. It is worth noting that much of lines-changed

検索: sony

10.375s 東京: pm4:52 Tuesday

- * **Most PS3 support got mainlined in 2.6.23**
- * **Become third contributed company in Japan**
 - both in # of change lines and change sets
- * **Who wrote 2.6.23**
 - <http://lwn.net/Articles/247582/>
- * **Who's writing Linux?**
 - <http://www.linuxworld.com/news/2007/092007-kernel.html>

Summary of Community Activities

- * **OSS for CBE ML**
 - Discussing Cell/B.E. open software, including PS3 Linux kernel, other Linux on Cell, libspe, and so on.
 - URL: <https://ozlabs.org/mailman/listinfo/cbe-oss-dev>
- * **Linux on PowerPC developers' ML**
 - URL: <https://ozlabs.org/mailman/listinfo/linuxppc-dev>
- * **GCC Web Site and ML**
 - URL: <http://gcc.gnu.org/>
- * **Binutils Web Site and ML**
 - URL: <http://sourceware.org/binutils/>
- * **GDB web Site and ML**
 - URL: <http://sourceware.org/gdb/>
- * **Newlib Web Site and ML**
 - The matter about newlib, which is used as C library for SPE, is discussed in Newlib ML.
 - URL: <http://sourceware.org/newlib/>
- * **kboot Web Site and ML**
 - URL: <http://kboot.sourceforge.net/>
- * **Perfmon2**
 - URL: <http://perfmon2.sourceforge.net>

Summary of Community Activities - 2

- * Bullet Physics Library
 - SIMD and Vector math library also included in
 - <http://sourceforge.net/projects/bullet/>

- * OpenCV for Cell
 - open source computer vision library
 - http://cell.fixstars.com/opencv/index.php/OpenCV_on_the_Cell

- * Cell Broadband Engine Architecture forum @ IBM developersWorks
 - http://www.ibm.com/developerworks/forums/dw_forum.jsp?forum=739&cat=46

- * Georgia Tech Cell BE Libraries
 - FFT, GZIP, MPEG2 and RC5
 - <http://sourceforge.net/projects/cellbuzz>

It's Time to Try By Yourself !!

- * Anyone can use favorite Linux Distro on PS3.
- * Utility of SPE is key of high performance.
- * Starting SPE programming is easy and must be fun.
- * Various community activities have arisen.

Let's Install Linux to your PS3 and
get start SPE programming.

Appendix

Development Packages

- * *IBM Cell BE SDK 3.0* - FC7 based packages
 - http://www.bsc.es/plantillaH.php?cat_id=459
- * Documents
 - http://www.bsc.es/plantillaH.php?cat_id=326
- * YDL 5.0.2 includes some of *IBM Cell BE SDK 2.0* packages
- * Ubuntu 7.04 –Beta deb packages (*IBM Cell BE SDK 2.1* based)
 - deb [http://people.ubuntu.com/~doko/ubuntu feisty-proposed/](http://people.ubuntu.com/~doko/ubuntu%20feisty-proposed/)
 - deb-src [http://people.ubuntu.com/~doko/ubuntu feisty-proposed/](http://people.ubuntu.com/~doko/ubuntu%20feisty-proposed/)
- * Ubuntu 7.10 will include some of *IBM Cell BE SDK 3.0* packages

Development Packages

- * **Mandatory**

- **GCC for SPU**

- * spu-gcc-4.1.1-107.ppc.rpm
 - * spu-gcc-c++-4.1.1-107.ppc.rpm

- **BINUTILS for SPU**

- * spu-binutils-2.17.50-8.33ppc.rpm

- **GDB for both PPU and SPU**

- * ppu-gdb-6.6.50-28.ppc.rpm

- **newlib - libc for SPU**

- * spu-newlib-1.15.0-82.ppc.rpm

- **libspe2 - SPE management library**

- * libspe2-2.2.0-91.ppc.rpm / .ppc64.rpm
 - * libspe2-devel-2.2.0-91.ppc.rpm / .ppc64.rpm
 - * libspe2man-2.2.0-91.noarch.rpm

- **elfspe - SPE elf launcher**

- * elfspe2-2.2.0-91.ppc.rpm

Development Packages

- * Recommended

- **PPU optimized toolchain**

- * **PPU BINUTILS**

- ppu-binutils-2.17.50-32.ppc.rpm

- * **PPU GCC**

- ppu-gcc-4.1.1-57.ppc.rpm

- ppu-gcc-c++-4.1.1-47.ppc.rpm

- ppu-gcc-fortran-4.1.1-57.ppc.rpm

- **SIMD and Vector Math**

- * simd math 1.02 & vector math 1.01

- * As part of bullet physics lib

- http://sourceforge.net/project/showfiles.php?group_id=147573

Cell/B.E. Programming Tips

- * Consider to use `ppu-gcc`.
 - 20% faster code, according with some measurement.
- * Be careful, default GCC ABI is up to configuration.
 - `ppu-gcc` 64bit (`-m64`) is default
 - `gcc` on Fedora 32bit (`-m32`) is default
- * Recommended `spu-gcc/spu-g++` options
 - `-Os -ffast-math -ftree-vectorize`
`-Wl,-gc-sections -ffunction-sections -fdata-sections`
`-fno-rtti -fno-exceptions (g++)`

Use `-O3 -funroll-all-loops`, instead of `-Os`, for faster code
- * Consider giving a hint to `spu-gcc` for branch prediction,
 - `builtin_expect()`

Summary of Community Activities

- * **OSS for CBE ML**
 - Discussing Cell/B.E. open software, including PS3 Linux kernel, other Linux on Cell, libspe, and so on.
 - URL: <https://ozlabs.org/mailman/listinfo/cbe-oss-dev>
- * **Linux on PowerPC developers' ML**
 - URL: <https://ozlabs.org/mailman/listinfo/linuxppc-dev>
- * **GCC Web Site and ML**
 - URL: <http://gcc.gnu.org/>
- * **Binutils Web Site and ML**
 - URL: <http://sourceware.org/binutils/>
- * **GDB web Site and ML**
 - URL: <http://sourceware.org/gdb/>
- * **Newlib Web Site and ML**
 - The matter about newlib, which is used as C library for SPE, is discussed in Newlib ML.
 - URL: <http://sourceware.org/newlib/>
- * **kboot Web Site and ML**
 - URL: <http://kboot.sourceforge.net/>
- * **Perfmon2**
 - URL: <http://perfmon2.sourceforge.net>

Summary of Community Activities - 2

- * Bullet Physics Library

- SIMD and Vector math library also included in
- <http://sourceforge.net/projects/bullet/>

- * OpenCV for Cell

- open source computer vision library
- http://cell.fixstars.com/opencv/index.php/OpenCV_on_the_Cell

- * Cell Broadband Engine Architecture forum @ IBM developersWorks

- http://www.ibm.com/developerworks/forums/dw_forum.jsp?forum=739&cat=46

- * Georgia Tech Cell BE Libraries

- FFT, GZIP, MPEG2 and RC5
- <http://sourceforge.net/projects/cellbuzz>

Legal Statement

- * My talk is based on public-available information and doesn't represent any of Sony Corp. and/or Sony Computer Entertainment Inc. positions and/or opinions.
- * “Linux” is a registered trademark of Linus Torvalds.
- * “PLAYSTATION” and “PS3” are registered trademarks and “Cell Broadband Engine” is a trademark of Sony Computer Entertainment Inc.
- * “IBM” and “IBM (logo)” are trademarks or registered trademarks of International Business Machines Corporation.
- * Other company, product, and service names/logs may be trademarks or service marks of others.